



**Devang Patel Institute of
Advance Technology and Research**
(A Constitute Institute of CHARUSAT)

Certificate

This is to certify that

Mr./Mrs. Prabln Bhagchandani
of Computer Engineering Class,
ID. No. 2ADCE006 has satisfactorily completed
his/ her term work in CE385 - Mobile Application Dev. for
the ending in Nov 2024/2025

Date :

14/10/24

Sign. of Faculty

Head of Department

PRACTICAL 1

AIM : 1.1 : Create “Custom Message” application. That will display “Custom Message” in the middle of the screen.

CODE:

XML FILE:

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
        xmlns:app="http://schemas.android.com/apk/res-auto"
        xmlns:tools="http://schemas.android.com/tools"
        android:id="@+id/main"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        tools:context=".MainActivity">

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Custom Message"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent" />

</androidx.constraintlayout.widget.ConstraintLayout>
```

MainActivity file:

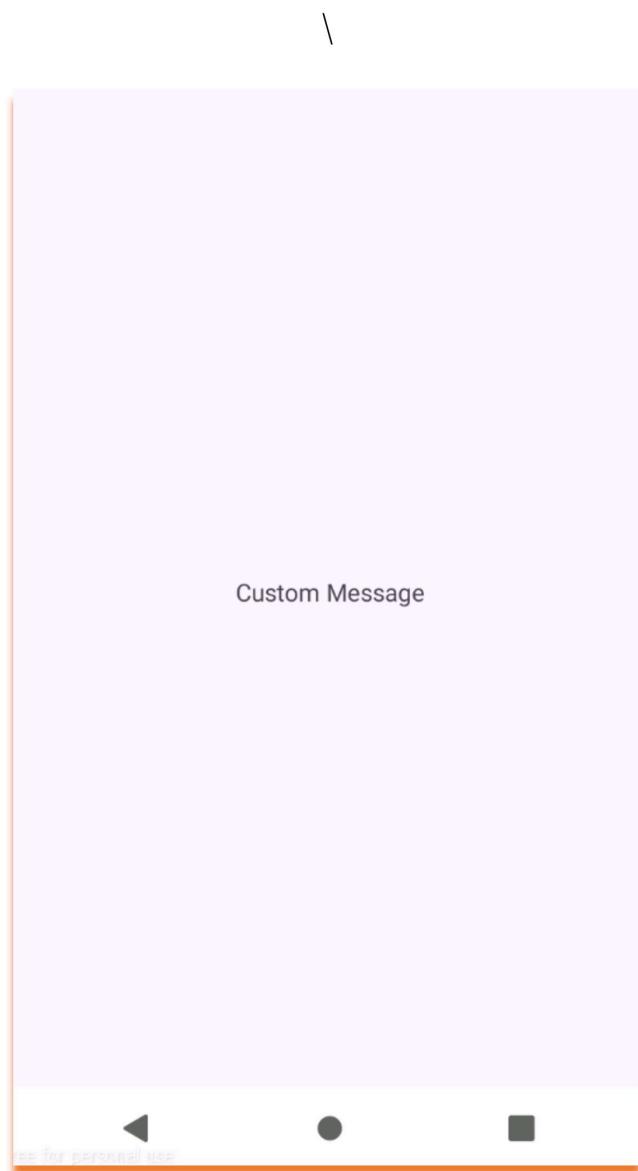
```
package com.example.new_practs_228

import android.os.Bundle
import androidx.activity.enableEdgeToEdge
import androidx.appcompat.app.AppCompatActivity
import android.core.view.ViewCompat
import android.core.view.WindowInsetsCompat

class MainActivity : AppCompatActivity() {
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        enableEdgeToEdge()
        setContentView(R.layout.activity_main)
```

```
ViewCompat.setOnApplyWindowInsetsListener(findViewById(R.id.main)) { v, insets ->
    val systemBars =
        insets.getInsets(WindowInsetsCompat.Type.systemBars())
    v.setPadding(systemBars.left, systemBars.top, systemBars.right,
        systemBars.bottom)
    insets
}
}
```

OUTPUT:



1.2: Design an application representing a simple calculator.

CODE:

XML FILE:

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
        xmlns:app="http://schemas.android.com/apk/res-auto"
        xmlns:tools="http://schemas.android.com/tools"
        android:id="@+id/main"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        tools:context=".MainActivity">

    <EditText
        android:id="@+id/editTextText2"
        android:layout_width="0dp"
        android:layout_height="wrap_content"
        android:hint="Enter first number"
        android:inputType="numberDecimal"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.5"
        app:layout_constraintWidth_default="spread"
        android:layout_marginTop="16dp"/>

    <EditText
        android:id="@+id/editTextText3"
        android:layout_width="0dp"
        android:layout_height="wrap_content"
        android:hint="Enter second number"
        android:inputType="numberDecimal"
        app:layout_constraintTop_toBottomOf="@+id/editTextText2"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        android:layout_marginTop="16dp"/>

    <TextView
        android:id="@+id/textView3"
        android:layout_width="0dp"
        android:layout_height="wrap_content"
        android:text="Result: "
        android:textSize="20sp"
        app:layout_constraintTop_toBottomOf="@+id/editTextText3"
```

```
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    android:layout_marginTop="16dp"/>
<Button
    android:id="@+id/button"
    android:layout_width="0dp"
    android:layout_height="wrap_content"
    android:text="Add"
    app:layout_constraintTop_toBottomOf="@+id/textView3"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    android:layout_marginTop="16dp"/>

<Button
    android:id="@+id/button2"
    android:layout_width="0dp"
    android:layout_height="wrap_content"
    android:text="Subtract"
    app:layout_constraintTop_toBottomOf="@+id/button"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    android:layout_marginTop="16dp"/>

<Button
    android:id="@+id/button3"
    android:layout_width="0dp"
    android:layout_height="wrap_content"
    android:text="Multiply"
    app:layout_constraintTop_toBottomOf="@+id/button2"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    android:layout_marginTop="16dp"/>

<Button
    android:id="@+id/button4"
    android:layout_width="0dp"
    android:layout_height="wrap_content"
    android:text="Divide"
    app:layout_constraintTop_toBottomOf="@+id/button3"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    android:layout_marginTop="16dp"/>
```

```

<Button
    android:id="@+id/button5"
    android:layout_width="0dp"
    android:layout_height="wrap_content"
    android:text="AC"
    app:layout_constraintTop_toBottomOf="@+id/button4"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    android:layout_marginTop="10dp"/>

<TextView
    android:id="@+id/textView"
    android:layout_width="0dp"
    android:layout_height="wrap_content"
    android:text="Log Display: "
    android:textSize="20sp"
    app:layout_constraintTop_toBottomOf="@+id/button5"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    android:layout_marginTop="16dp"/>

</androidx.constraintlayout.widget.ConstraintLayout>

```

MainActivity file:

```

package com.example.basic_calc_kotlin

import android.os.Bundle
import android.widget.Button
import android.widget.TextView
import android.widget.EditText
import androidx.activity.enableEdgeToEdge
import androidx.appcompat.app.AppCompatActivity
import androidx.core.view.ViewCompat
import androidx.core.view.WindowInsetsCompat

class MainActivity : AppCompatActivity() {
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        enableEdgeToEdge()
        setContentView(R.layout.activity_main)
        val etFirstNumber = findViewById<EditText>(R.id.editTextText2)
        val etSecondNumber =
            findViewById<EditText>(R.id.editTextText3)
    }
}

```

```
val tvResult = findViewById<TextView>(R.id.textView3)

val btnAdd = findViewById<Button>(R.id.button)
val btnSubtract = findViewById<Button>(R.id.button2)
val btnMultiply = findViewById<Button>(R.id.button3)
val btnDivide = findViewById<Button>(R.id.button4)
val btnac = findViewById<Button>(R.id.button5)

// Add
btnAdd.setOnClickListener {
    val num1 = etFirstNumber.text.toString().toDoubleOrNull()
    val num2 = etSecondNumber.text.toString().toDoubleOrNull()
    if (num1 != null && num2 != null) {
        val result = num1 + num2
        tvResult.text = "Result: $result"
    } else {
        tvResult.text = "Please enter valid numbers"
    }
}
// Subtract
btnSubtract.setOnClickListener {
    val num1 = etFirstNumber.text.toString().toDoubleOrNull()
    val num2 = etSecondNumber.text.toString().toDoubleOrNull()
    if (num1 != null && num2 != null) {
        val result = num1 - num2
        tvResult.text = "Result: $result"
    } else {
        tvResult.text = "Please enter valid numbers"
    }
}

// Multiply
btnMultiply.setOnClickListener {
    val num1 = etFirstNumber.text.toString().toDoubleOrNull()
    val num2 = etSecondNumber.text.toString().toDoubleOrNull()
    if (num1 != null && num2 != null) {
        val result = num1 * num2
        tvResult.text = "Result: $result"
    } else {
        tvResult.text = "Please enter valid numbers"
    }
}
```

```

// Divide
btnDivide.setOnClickListener {
    val num1 = etFirstNumber.text.toString().toDoubleOrNull()
    val num2 = etSecondNumber.text.toString().toDoubleOrNull()
    if (num1 != null && num2 != null) {
        if (num2 != 0.0) {
            val result = num1 / num2
            tvResult.text = "Result: $result"
        } else {
            tvResult.text = "Cannot divide by zero"
        }
    } else {
        tvResult.text = "Please enter valid numbers"
    }
}
//ac
btnac.setOnClickListener {

}
//activity life cycle

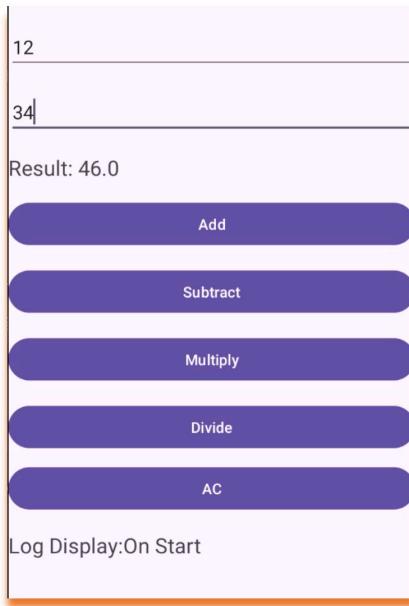
```

```

ViewCompat.setOnApplyWindowInsetsListener(findViewById(R.id.main)) { v, insets ->
    val systemBars =
    insets.getInsets(WindowInsetsCompat.Type.systemBars())
    v.setPadding(systemBars.left, systemBars.top, systemBars.right,
    systemBars.bottom)
    insets
}
}
}

```

OUTPUT:



1.3: Create a login application with following features:

1. Successful Login message in TextView with Green background if Username & password is correct
2. Failure message in TextView with Red background if Username or password is incorrect.
3. Disable login Button after three wrong login attempts.
4. Close application if user selects Cancel Button.

CODE:

XML FILE:

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:id="@+id/main"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".MainActivity">
    <!-- User name -->
    <EditText
        android:id="@+id/editTextText"
        android:layout_width="0dp"
        android:layout_height="wrap_content"
        android:hint="Enter Username"
        app:layout_constraintTop_toTopOf="parent"
```

```
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    android:inputType="text"
    android:layout_marginTop="16dp" />
<!-- password -->
<EditText
    android:id="@+id/editTextText2"
    android:layout_width="0dp"
    android:layout_height="wrap_content"
    android:hint="Enter Password"
    android:inputType="textPassword"
    app:layout_constraintTop_toBottomOf="@+id/editTextText"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    android:layout_marginTop="16dp" />
<!-- login button -->
<Button
    android:id="@+id/button"
    android:layout_width="0dp"
    android:layout_height="wrap_content"
    android:text="Login"
    app:layout_constraintTop_toBottomOf="@+id/editTextText2"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    android:layout_marginTop="16dp" />
<!-- Cancel button -->
<Button
    android:id="@+id/button2"
    android:layout_width="0dp"
    android:layout_height="wrap_content"
    android:text="Cancel"
    app:layout_constraintTop_toBottomOf="@+id/button"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    android:layout_marginTop="16dp" />
<!-- Result -->
<TextView
    android:id="@+id/textView"
    android:layout_width="0dp"
    android:layout_height="wrap_content"
    android:text=""
    android:textSize="20sp"
    android:gravity="center"
```

```
        android:visibility="gone"
        app:layout_constraintTop_toBottomOf="@+id/button2"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        android:layout_marginTop="16dp" />
</androidx.constraintlayout.widget.ConstraintLayout>
```

MainActivity file:

```
package com.example.p1_3_login

import android.graphics.Color
import android.os.Bundle
import android.widget.Button
import android.widget.EditText
import android.widget.TextView
import androidx.activity.enableEdgeToEdge
import androidx.appcompat.app.AppCompatActivity
import androidx.core.view.ViewCompat
import androidx.core.view.WindowInsetsCompat
import kotlin.system.exitProcess

class MainActivity : AppCompatActivity() {
    private var attempts = 0
    private val maxAttempts = 3
    private val correctUsername = "admin"
    private val correctPassword = "password123"
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        enableEdgeToEdge()
        setContentView(R.layout.activity_main)
        val etUsername = findViewById<EditText>(R.id.editTextText)
        val etPassword = findViewById<EditText>(R.id.editTextText2)
        val btnLogin = findViewById<Button>(R.id.button)
        val btnCancel = findViewById<Button>(R.id.button2)
        val tvResult = findViewById<TextView>(R.id.textView)
        btnLogin.setOnClickListener {
            val username = etUsername.text.toString()
            val password = etPassword.text.toString()
            // Check if login is successful
            if (username == correctUsername && password ==
                correctPassword) {
                tvResult.text = "Login Successful!"
                tvResult.setBackgroundColor(Color.GREEN)
            }
        }
    }
}
```

```

tvResult.setTextColor(Color.WHITE)
tvResult.setVisibility = TextView.VISIBLE
btnLogin.isEnabled = false
} else {
    attempts++
    if (attempts >= maxAttempts) {
        btnLogin.isEnabled = false
        tvResult.text = "Login Disabled! Too many attempts."
        tvResult.setBackgroundColor(Color.RED)
        tvResult.setTextColor(Color.WHITE)
    } else {
        tvResult.text = "Login Failed! Attempts remaining:
${maxAttempts - attempts}"
        tvResult.setBackgroundColor(Color.RED)
        tvResult.setTextColor(Color.WHITE)
    }
    tvResult.setVisibility = TextView.VISIBLE
}
}
btnCancel.setOnClickListener {
    // Close the application when Cancel button is pressed
    finishAffinity()
    exitProcess(0)
}

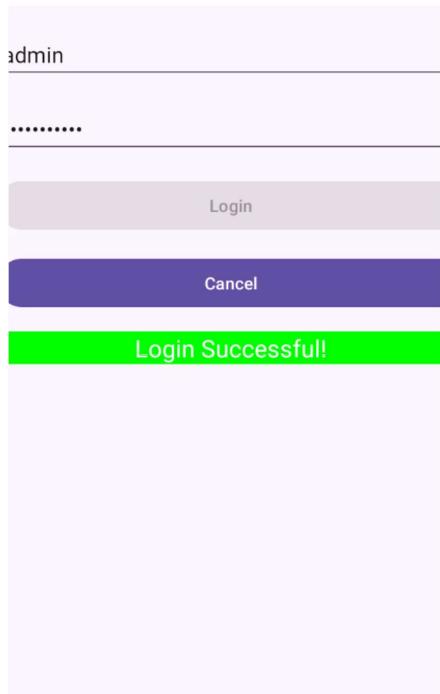
```

```

ViewCompat.setOnApplyWindowInsetsListener(findViewById(R.id.main)) { v, insets ->
    val systemBars =
    insets.getInsets(WindowInsetsCompat.Type.systemBars())
    v.setPadding(systemBars.left, systemBars.top, systemBars.right,
    systemBars.bottom)
    insets
}
}

```

OUTPUT:



1.4: Create android application to demonstrate life cycle of activity.

CODE:

XML FILE:

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:id="@+id/main"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".MainActivity">
```

```
<TextView
    android:id="@+id/textView"
    android:layout_width="0dp"
    android:layout_height="wrap_content"
    android:text="Lifecycle Logs"
    android:textSize="18sp"
    android:gravity="center"
```

```
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    android:layout_marginTop="16dp" />
</androidx.constraintlayout.widget.ConstraintLayout>
```

MainActivity file:

```
package com.example.activity_lifecycle
```

```
import android.os.Bundle
import android.util.Log
import android.widget.TextView
import androidx.activity.enableEdgeToEdge
import androidx.appcompat.app.AppCompatActivity
import androidx.core.view.ViewCompat
import androidx.core.view.WindowInsetsCompat

class MainActivity : AppCompatActivity() {
    private lateinit var tvLifecycle: TextView
    // Tag for logging
    private val TAG = "ActivityLifecycle"
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        enableEdgeToEdge()
        setContentView(R.layout.activity_main)
        tvLifecycle = findViewById(R.id.textView)

        // Log and display onCreate state
        logAndDisplay("onCreate")
    }

    override fun onStart() {
        super.onStart()
        // Log and display onStart state
        logAndDisplay("onStart")
    }

    override fun onResume() {
        super.onResume()
        // Log and display onResume state
        logAndDisplay("onResume")
    }

    override fun onPause() {
```

```
super.onPause()
// Log and display onPause state
logAndDisplay("onPause")
}

override fun onStop() {
    super.onStop()
    // Log and display onStop state
    logAndDisplay("onStop")
}

override fun onRestart() {
    super.onRestart()
    // Log and display onRestart state
    logAndDisplay("onRestart")
}

override fun onDestroy() {
    super.onDestroy()
    // Log and display onDestroy state
    logAndDisplay("onDestroy")
}

// Method to log and update the TextView with the current lifecycle
state
private fun logAndDisplay(state: String) {
    Log.d(TAG, state)
    tvLifecycle.text = "Current state: $state"
}

}
```

OUTPUT:

```
2024-10-07 10:45:57.632 2958-2994 ProfileInstaller com.example.activity_lifecycle
2024-10-07 10:46:01.222 2958-2958 ActivityLifecycle com.example.activity_lifecycle
2024-10-07 10:46:01.710 2958-2958 ActivityLifecycle com.example.activity_lifecycle
2024-10-07 10:46:04.544 2958-2958 ActivityLifecycle com.example.activity_lifecycle
2024-10-07 10:46:04.545 2958-2958 ActivityLifecycle com.example.activity_lifecycle
2024-10-07 10:46:04.546 2958-2958 ActivityLifecycle com.example.activity_lifecycle
2024-10-07 10:46:04.586 2958-2980 OpenGLRenderer com.example.activity_lifecycle
2024-10-07 10:46:11.996 2958-2971 System com.example.activity_lifecycle
2024-10-07 10:46:52.593 2958-2958 ActivityLifecycle com.example.activity_lifecycle
2024-10-07 10:46:53.206 2958-2958 ActivityLifecycle com.example.activity_lifecycle
2024-10-07 10:46:59.427 2958-2958 ActivityLifecycle com.example.activity_lifecycle
2024-10-07 10:46:59.430 2958-2958 ActivityLifecycle com.example.activity_lifecycle
2024-10-07 10:46:59.436 2958-2958 ActivityLifecycle com.example.activity_lifecycle
2024-10-07 10:46:59.473 2958-2980 OpenGLRenderer com.example.activity_lifecycle
2024-10-07 10:47:06.982 2958-2958 ActivityLifecycle com.example.activity_lifecycle
2024-10-07 10:47:07.450 2958-2958 ActivityLifecycle com.example.activity_lifecycle
2024-10-07 10:47:11.052 2958-2958 ActivityLifecycle com.example.activity_lifecycle
2024-10-07 10:47:11.054 2958-2958 ActivityLifecycle com.example.activity_lifecycle
2024-10-07 10:47:11.056 2958-2958 ActivityLifecycle com.example.activity_lifecycle
2024-10-07 10:47:11.101 2958-2980 OpenGLRenderer com.example.activity_lifecycle
D Installing profile for com.example.acti...
D onPause
D onStop
D onRestart
D onStart
D onResume
E Unable to match the desired swap behav...
W A resource failed to call close.
D onPause
D onStop
D onRestart
D onStart
D onResume
E Unable to match the desired swap behav...
D onPause
D onStop
D onRestart
D onStart
D onResume
E Unable to match the desired swap behav...
D onPause
D onStop
D onRestart
D onStart
D onResume
E Unable to match the desired swap behav...
```

PRACTICAL 2

AIM : Create application which add 10 items in list using listview.

CODE:

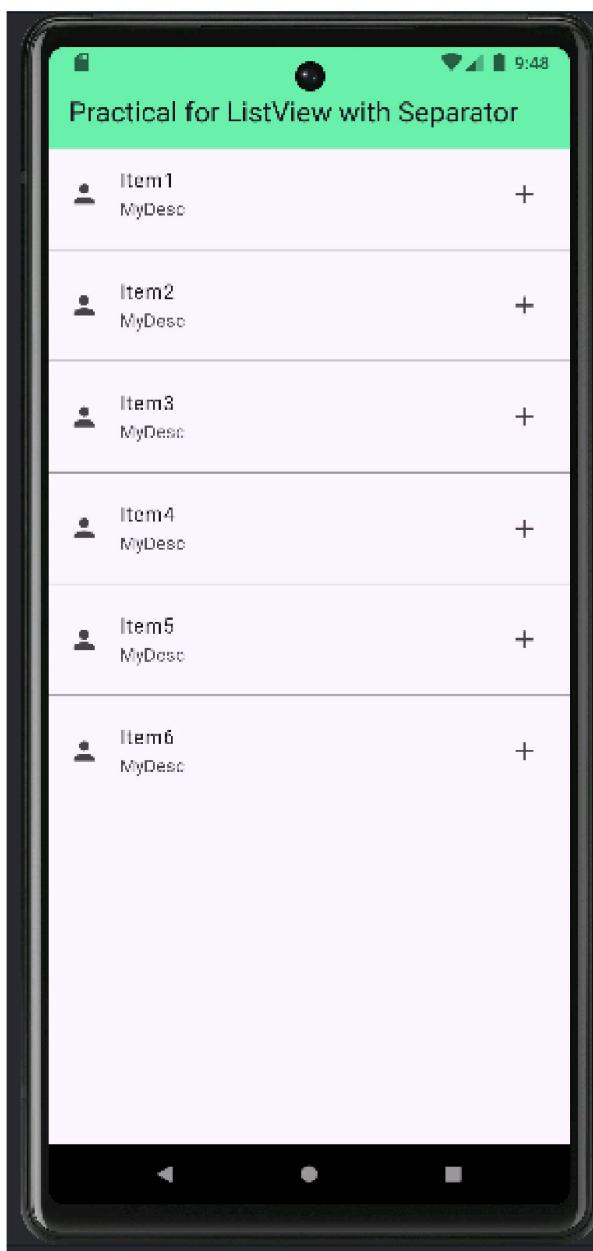
XML FILE:

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:orientation="vertical">
    <!-- ListView-->
    <ListView
        android:id="@+id/listView"
        android:layout_width="match_parent"
        android:layout_height="wrap_content" />
</LinearLayout>
```

MainActivity file:

```
package com.example.p2_listview // Ensure this matches your package
name
import android.os.Bundle
import android.widget.ArrayAdapter
import android.widget.ListView
import androidx.appcompat.app.AppCompatActivity
class MainActivity : AppCompatActivity() {
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_main) // This references the XML
layout
        // Find the ListView by its ID
        val listView: ListView = findViewById(R.id.listView)
        // Create a list of
        val df = listOf("Item1", "Item2", "Item3", "Item4", "Item5",
"Item6")
        // Bind the list of to the ListView using ArrayAdapter
        val adapter = ArrayAdapter(this,
            android.R.layout.simple_list_item_1, df)
        listView.adapter = adapter
    }
}
```

OUTPUT:



Practical – 3

AIM: Implement the given layout using Linear Layout.

CODE:

ActivityMain:

```
package com.example.prac3

import android.os.Bundle

import androidx.activity.enableEdgeToEdge

import androidx.appcompat.app.AppCompatActivity

import androidx.core.view.ViewCompat

import androidx.core.view.WindowInsetsCompat

class MainActivity : AppCompatActivity() {

    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        enableEdgeToEdge()
        setContentView(R.layout.activity_main)

        ViewCompat.setOnApplyWindowInsetsListener(findViewById(R.id.main)) { v,
        insets ->

            val systemBars = insets.getInsets(WindowInsetsCompat.Type.systemBars())

            v.setPadding(systemBars.left, systemBars.top, systemBars.right,
            systemBars.bottom)

            insets
        }
    }
}
```

XML FILE:

```
<?xml version="1.0" encoding="utf-8"?>

<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
```

```
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    android:padding="8dp">

    <!-- Top Half -->
    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="0dp"
        android:layout_weight="1"
        android:orientation="horizontal">

        <!-- Block 1 (Spanning two rows) -->
        <TextView
            android:layout_width="0dp"
            android:layout_height="match_parent"
            android:layout_weight="2"
            android:layout_gravity="center"
            android:background="#E91E63"
            android:gravity="center"
            android:text="1"
            android:textSize="24sp"
            android:textColor="#FFFFFF" />

        <!-- Right-side blocks (Block 2, 3, 4) -->
        <LinearLayout
            android:layout_width="0dp"
            android:layout_height="match_parent"
```

```
    android:layout_weight="2"
    android:orientation="vertical">

<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="0dp"
    android:layout_weight="1"
    android:orientation="horizontal">

    <!-- Block 2 -->
    <TextView
        android:layout_width="0dp"
        android:layout_height="match_parent"
        android:layout_weight="1"
        android:background="#4CAF50"
        android:gravity="center"
        android:text="2"
        android:textSize="24sp"
        android:textColor="#FFFFFF" />

    <!-- Block 3 -->
    <TextView
        android:layout_width="0dp"
        android:layout_height="match_parent"
        android:layout_weight="1"
        android:background="#FF9800"
        android:gravity="center"
        android:text="3"
```

```
        android:textSize="24sp"
        android:textColor="#FFFFFF" />
    </LinearLayout>

    <!-- Block 4 -->
    <TextView
        android:layout_width="match_parent"
        android:layout_height="0dp"
        android:layout_weight="1"
        android:background="#F44336"
        android:gravity="center"
        android:text="4"
        android:textSize="24sp"
        android:textColor="#FFFFFF" />
    </LinearLayout>
</LinearLayout>

<!-- Bottom Half -->
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="0dp"
    android:layout_weight="1"
    android:orientation="horizontal">

    <!-- Left blocks (Block 5, 6) -->
    <LinearLayout
        android:layout_width="0dp"
        android:layout_height="match_parent"
```

```
    android:layout_weight="2"
    android:orientation="vertical">

    <!-- Block 5 -->
    <TextView
        android:layout_width="match_parent"
        android:layout_height="0dp"
        android:layout_weight="1"
        android:background="#9C27B0"
        android:gravity="center"
        android:text="5"
        android:textSize="24sp"
        android:textColor="#FFFFFF" />

    <!-- Block 6 -->
    <TextView
        android:layout_width="match_parent"
        android:layout_height="0dp"
        android:layout_weight="1"
        android:background="#03A9F4"
        android:gravity="center"
        android:text="6"
        android:textSize="24sp"
        android:textColor="#FFFFFF" />

</LinearLayout>

    <!-- Block 1 (Repeating block) -->
    <TextView
```

```
    android:layout_width="0dp"
    android:layout_height="match_parent"
    android:layout_weight="2"
    android:background="#E91E63"
    android:gravity="center"
    android:text="1"
    android:textSize="24sp"
    android:textColor="#FFFFFF" />

</LinearLayout>
```

```
<!-- Bottom Row -->

<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="0dp"
    android:layout_weight="1"
    android:orientation="horizontal">
```

```
<!-- Block 2 -->

<TextView
    android:layout_width="0dp"
    android:layout_height="match_parent"
    android:layout_weight="1"
    android:background="#4CAF50"
    android:gravity="center"
    android:text="2"
    android:textSize="24sp"
    android:textColor="#FFFFFF" />
```

```
<!-- Block 3 -->
<TextView
    android:layout_width="0dp"
    android:layout_height="match_parent"
    android:layout_weight="1"
    android:background="#FF9800"
    android:gravity="center"
    android:text="3"
    android:textSize="24sp"
    android:textColor="#FFFFFF" />

<!-- Block 4 -->
<TextView
    android:layout_width="0dp"
    android:layout_height="match_parent"
    android:layout_weight="1"
    android:background="#F44336"
    android:gravity="center"
    android:text="4"
    android:textSize="24sp"
    android:textColor="#FFFFFF" />

</LinearLayout>

</LinearLayout>
```

OUTPUT:



PRACTICAL 4

AIM : Create an application in which user pass the data with intent object to another activity.

CODE:

```
package com.example.myapplication

import android.content.Intent
import android.os.Bundle
import android.view.View
import android.widget.Button
import android.widget.EditText
import android.widget.TextView
import androidx.appcompat.app.AppCompatActivity
import androidx.constraintlayout.widget.ConstraintLayout
import androidx.constraintlayout.widget.ConstraintSet
import android.view.ViewGroup.LayoutParams

class MainActivity : AppCompatActivity() {

    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)

        // Create the layout programmatically
        val layout = ConstraintLayout(this)
        layout.layoutParams = ConstraintLayout.LayoutParams(
            LayoutParams.MATCH_PARENT,
            LayoutParams.MATCH_PARENT
        )

        // EditText for user input
        val inputField = EditText(this)
        inputField.id = View.generateViewId()
        inputField.hint = "Enter something"
        layout.addView(inputField)

        // Button to send data
        val sendButton = Button(this)
        sendButton.id = View.generateViewId()
        sendButton.text = "Send"
        layout.addView(sendButton)

        // TextView to display received data (in case we handle
        // receiving in the same activity)
        val displayTextView = TextView(this)
        displayTextView.id = View.generateViewId()
        displayTextView.text = "Received data will appear here"
        layout.addView(displayTextView)

        // Define constraints for the layout elements
        val set = ConstraintSet()
        set.clone(layout)
        set.connect(inputField.id, ConstraintSet.TOP, layout.id,
        ConstraintSet.TOP, 100)
        set.connect(inputField.id, ConstraintSet.LEFT, layout.id,
        ConstraintSet.LEFT, 16)
        set.connect(inputField.id, ConstraintSet.RIGHT, layout.id,
        ConstraintSet.RIGHT, 16)
    }
}
```

```

        set.connect(sendButton.id, ConstraintSet.TOP, inputField.id,
ConstraintSet.BOTTOM, 16)
        set.connect(sendButton.id, ConstraintSet.LEFT, layout.id,
ConstraintSet.LEFT, 16)
        set.connect(sendButton.id, ConstraintSet.RIGHT, layout.id,
ConstraintSet.RIGHT, 16)

        set.connect(displayTextView.id, ConstraintSet.TOP,
sendButton.id, ConstraintSet.BOTTOM, 32)
        set.connect(displayTextView.id, ConstraintSet.LEFT,
layout.id, ConstraintSet.LEFT, 16)
        set.connect(displayTextView.id, ConstraintSet.RIGHT,
layout.id, ConstraintSet.RIGHT, 16)

        set.applyTo(layout)

    // Set the layout as the content view
    setContentView(layout)

    // Send data to the second activity
    sendButton.setOnClickListener {
        val userInput = inputField.text.toString()

        // Create an Intent to start the second activity
        val intent = Intent(this, SecondActivity::class.java)
        // Add the data to the intent
        intent.putExtra("user_input", userInput)

        // Start the second activity
        startActivity(intent)
    }
}

// SecondActivity that receives the data
class SecondActivity : AppCompatActivity() {

    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)

        // Create the layout programmatically
        val layout = ConstraintLayout(this)
        layout.layoutParams = ConstraintLayout.LayoutParams(
            LayoutParams.MATCH_PARENT,
            LayoutParams.MATCH_PARENT
        )

        // TextView to display received data
        val displayTextView = TextView(this)
        displayTextView.id = View.generateViewId()
        layout.addView(displayTextView)

        // Define constraints for the TextView
        val set = ConstraintSet()
        set.clone(layout)
        set.connect(displayTextView.id, ConstraintSet.TOP, layout.id,
ConstraintSet.TOP, 100)
        set.connect(displayTextView.id, ConstraintSet.LEFT,
layout.id, ConstraintSet.LEFT, 16)
        set.connect(displayTextView.id, ConstraintSet.RIGHT,

```

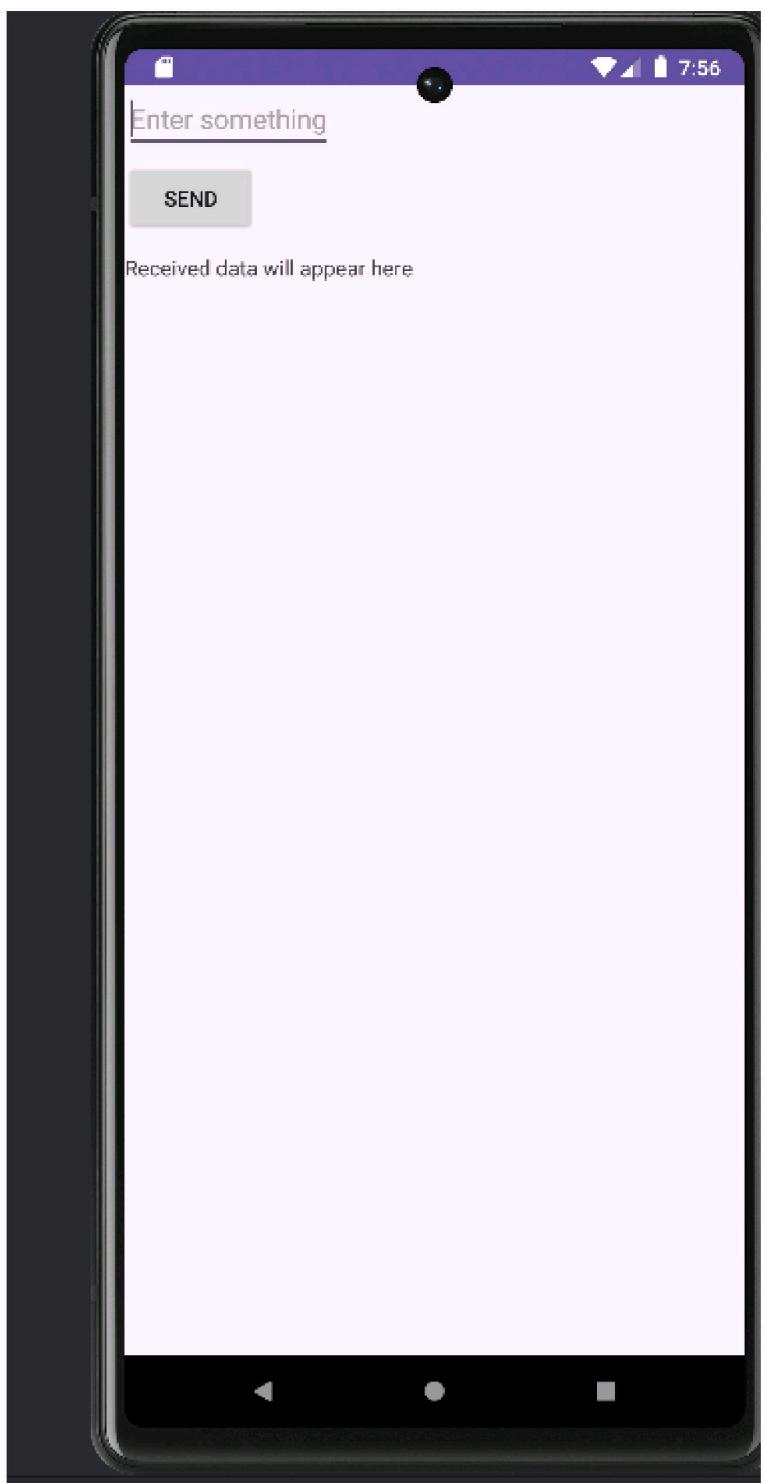
```
layout.id, ConstraintSet.RIGHT, 16)
set.applyTo(layout)

// Set the layout as the content view
setContentView(layout)

// Get the data passed from MainActivity
val receivedData = intent.getStringExtra("user_input")

// Display the data in the TextView
displayTextView.text = receivedData
}
}
```

OUTPUT:



Practical-5

Prac – 5 (A) (Custom message)

Code:

```
import 'package:flutter/material.dart';

void main() {
  runApp(const CustomMessageApp());
}

class CustomMessageApp extends StatelessWidget {
  const CustomMessageApp({Key? key}) : super(key: key);

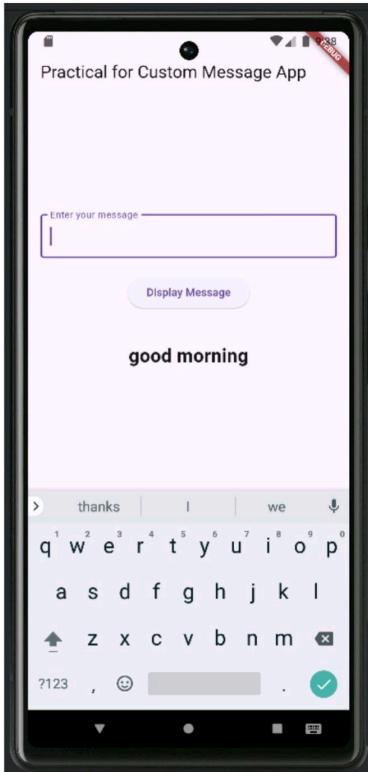
  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      title: 'Practical for Custom Message App',
      theme: ThemeData(
        primarySwatch: Colors.purple,
      ),
      home: const CustomMessageScreen(),
    );
  }
}

class CustomMessageScreen extends StatefulWidget {
  const CustomMessageScreen({Key? key}) : super(key: key);

  @override
  _CustomMessageScreenState createState() => _CustomMessageScreenState();
}
```

```
class _CustomMessageScreenState extends State<CustomMessageScreen> {  
  String _message = 'The message will come here';  
  final TextEditingController _controller = TextEditingController();  
  
  void _updateMessage() {  
    setState(() {  
      _message = _controller.text;  
    });  
    _controller.clear(); // Clear the text field after submitting  
  }  
  
  @override  
  Widget build(BuildContext context) {  
    return Scaffold(  
      appBar: AppBar(  
        title: const Text('Practical for Custom Message App'),  
      ),  
      body: Padding(  
        padding: const EdgeInsets.all(16.0),  
        child: Column(  
          mainAxisAlignment: MainAxisAlignment.center,  
          children: [  
            TextField(  
              controller: _controller,  
              decoration: const InputDecoration(  
                border: OutlineInputBorder(),  
                labelText: 'Enter your message',  
              ),  
            ),  
            const SizedBox(height: 20),  
          ],  
        ),  
      ),  
    );  
  }  
}
```

```
ElevatedButton(  
    onPressed: _updateMessage,  
    child: const Text('Display Message'),  
,  
const SizedBox(height: 40),  
Text(  
    _message,  
    style: const TextStyle(  
        fontSize: 24,  
        fontWeight: FontWeight.bold,  
,  
    textAlign: TextAlign.center,  
,  
],  
,  
,  
);  
}  
}
```



Prac – 5 (B) (Simple Calculator)

Code:

```
import 'package:flutter/material.dart';

void main() {
  runApp(const CalculatorApp());
}

class CalculatorApp extends StatelessWidget {
  const CalculatorApp({Key? key}) : super(key: key);

  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      title: 'Practical for Simple Calculator',
      theme: ThemeData(
```

```
    primarySwatch: Colors.purple,  
),  
home: const CalculatorScreen(),  
);  
}  
}  
  
class CalculatorScreen extends StatefulWidget {  
const CalculatorScreen({Key? key}) : super(key: key);  
  
@override  
_CalculatorScreenState createState() => _CalculatorScreenState();  
}  
  
class _CalculatorScreenState extends State<CalculatorScreen> {  
String _output = '0';  
String _operand = " ";  
double _num1 = 0;  
double _num2 = 0;  
  
void _buttonPressed(String buttonText) {  
setState(() {  
if (buttonText == 'C') {  
_output = '0';  
_num1 = 0;  
_num2 = 0;  
_operand = " ";  
} else if (buttonText == '+' || buttonText == '-' || buttonText == '×' || buttonText == '÷') {  
_num1 = double.parse(_output);  
_operand = buttonText;  
_output = '0';
```

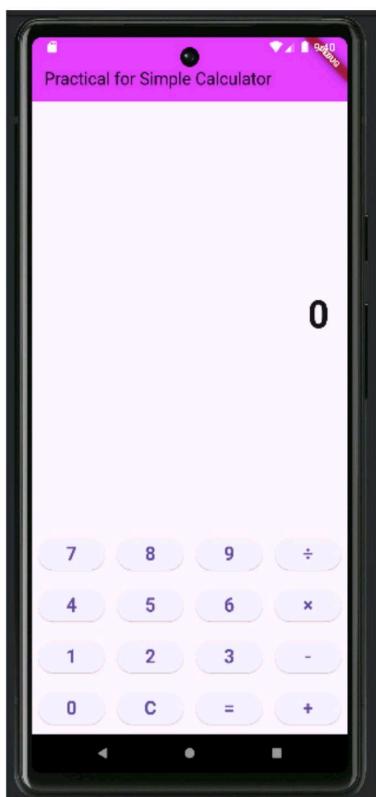
```
        } else if (buttonText == '=') {  
            _num2 = double.parse(_output);  
            if (_operand == '+') {  
                _output = (_num1 + _num2).toString();  
            } else if (_operand == '-') {  
                _output = (_num1 - _num2).toString();  
            } else if (_operand == '×') {  
                _output = (_num1 * _num2).toString();  
            } else if (_operand == '÷') {  
                _output = (_num1 / _num2).toString();  
            }  
            _num1 = 0;  
            _num2 = 0;  
            _operand = "=";  
        } else {  
            if (_output == '0') {  
                _output = buttonText;  
            } else {  
                _output += buttonText;  
            }  
        }  
    });  
}  
  
Widget _buildButton(String buttonText) {  
    return Expanded(  
        child: Padding(  
            padding: const EdgeInsets.all(8.0),  
            child: ElevatedButton(  
                onPressed: () => _buttonPressed(buttonText),  
                child: Text(  
                   
```

```
                    
```

```
buttonText,  
    style: const TextStyle(  
        fontSize: 24.0,  
        fontWeight: FontWeight.bold,  
    ),  
    ),  
),  
);  
}  
  
@override  
Widget build(BuildContext context) {  
    return Scaffold(  
        appBar: AppBar(  
            title: const Text('Practical for Simple Calculator'),  
            backgroundColor: Colors.purpleAccent,  
        ),  
        body: Column(  
            children: <Widget>[  
                Expanded(  
                    child: Container(  
                        padding: const EdgeInsets.all(24.0),  
                        alignment: Alignment.centerRight,  
                        child: Text(  
                            _output,  
                            style: const TextStyle(  
                                fontSize: 48.0,  
                                fontWeight: FontWeight.bold,  
                            ),  
                        ),  
                ),
```

```
        ),  
        ),  
        Column(  
            children: <Widget>[  
                Row(  
                    children: <Widget>[  
                        _buildButton('7'),  
                        _buildButton('8'),  
                        _buildButton('9'),  
                        _buildButton('÷'),  
                    ],  
                ),  
                Row(  
                    children: <Widget>[  
                        _buildButton('4'),  
                        _buildButton('5'),  
                        _buildButton('6'),  
                        _buildButton('×'),  
                    ],  
                ),  
                Row(  
                    children: <Widget>[  
                        _buildButton('1'),  
                        _buildButton('2'),  
                        _buildButton('3'),  
                        _buildButton('‐'),  
                    ],  
                ),  
                Row(  
                    children: <Widget>[  
                        _buildButton('0'),
```

```
_buildButton('C'),  
_buildButton('=')  
_buildButton('+'),  
],  
,  
],  
,  
],  
,  
),  
);  
}  
}
```



Practical-6

Part – 6 (ListView)

Code:

```
import 'package:flutter/material.dart';

void main() {
  runApp(const ListViewApp());
}

class ListViewApp extends StatelessWidget {
  const ListViewApp({Key? key}) : super(key: key);

  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      title: 'Practical for ListView with Separator',
      theme: ThemeData(
```

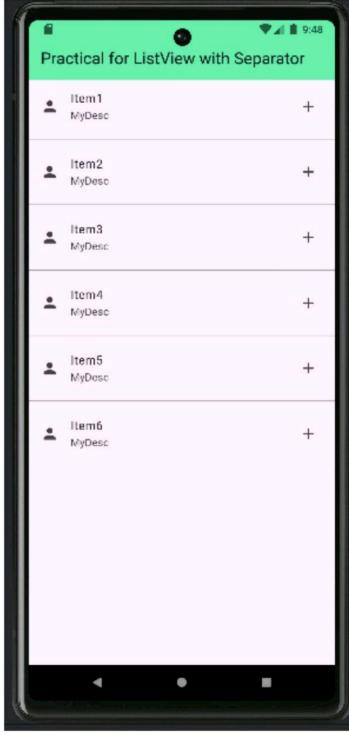
```
    primarySwatch: Colors.blue,  
),  
color: Colors.greenAccent,  
debugShowCheckedModeBanner: false,  
home: const ListViewScreen(),  
);  
}  
}
```

```
class ListViewScreen extends StatelessWidget {  
const ListViewScreen({Key? key}) : super(key: key);  
  
@override  
Widget build(BuildContext context) {  
// List of items (with leading, title, subtitle, trailing)  
final List<Map<String, dynamic>> items = [  
{  
'leading': Icons.person,  
'title': 'Item1',  
'subtitle': 'MyDesc',  
'trailing': Icons.add,  
},  
{  
'leading': Icons.person,  
'title': 'Item2',  
'subtitle': 'MyDesc',  
'trailing': Icons.add,  
},  
{  
'leading': Icons.person,  
'title': 'Item3',
```

```
'subtitle': 'MyDesc',
'trailing': Icons.add,
},
{
'leading': Icons.person,
'title': 'Item4',
'subtitle': 'MyDesc',
'trailing': Icons.add,
},
{
'leading': Icons.person,
'title': 'Item5',
'subtitle': 'MyDesc',
'trailing': Icons.add,
},
{
'leading': Icons.person,
'title': 'Item6',
'subtitle': 'MyDesc',
'trailing': Icons.add,
},
],
];
```

```
return Scaffold(
  appBar: AppBar(
    title: const Text('Practical for ListView with Separator'),
    backgroundColor: Colors.greenAccent,
  ),
  body: ListView.separated(
    itemCount: items.length,
    itemBuilder: (context, index) {
```

```
return ListTile(  
    leading: Icon(items[index]['leading']),  
    title: Text(items[index]['title']),  
    subtitle: Text(items[index]['subtitle']),  
    trailing: Icon(items[index]['trailing']),  
    onTap: () {  
        // Handle item tap if needed  
        print('Clicked and Tapped on ${items[index]['title']}');  
    },  
);  
,  
},  
separatorBuilder: (context, index) {  
    return const Divider(  
        color: Colors.grey,  
        thickness: 1.0,  
    );  
},  
,  
);  
}  
}
```



Practical-7

Part – 7 (Grid Layout)

Code:

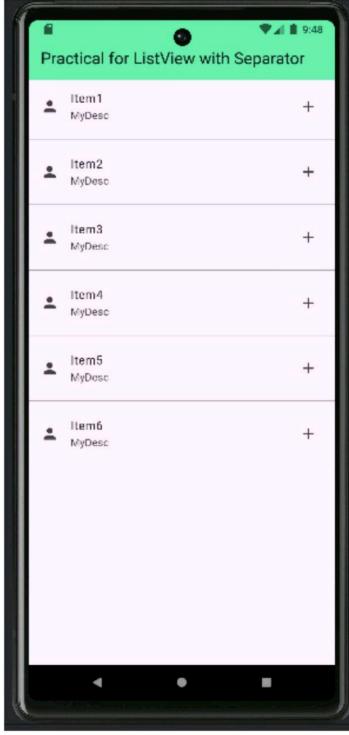
```
import 'package:flutter/material.dart';

void main() {
  runApp(const LayoutApp());
}

class LayoutApp extends StatelessWidget {
  const LayoutApp({Key? key}) : super(key: key);

  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      title: 'Custom Layout',

```



Practical-7

Part – 7 (Grid Layout)

Code:

```
import 'package:flutter/material.dart';

void main() {
  runApp(const LayoutApp());
}

class LayoutApp extends StatelessWidget {
  const LayoutApp({Key? key}) : super(key: key);

  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      title: 'Custom Layout',

```

```
home: const LayoutScreen(),  
);  
}  
}  
  
class LayoutScreen extends StatelessWidget {  
const LayoutScreen({Key? key}) : super(key: key);  
  
@override  
Widget build(BuildContext context) {  
return Scaffold(  
appBar: AppBar(  
title: const Text('Practical for Layout'),  
,  
body: Column(  
children: [  
// First half of the layout  
Expanded(  
child: Row(  
children: [  
// Large box on the left  
Expanded(  
flex: 1,  
child: Container(  
color: Colors.pinkAccent,  
child: const Center(  
child: Text(  
'I',  
style: TextStyle(  
fontSize: 48,  
color: Colors.white,
```



```
'3',  
    style: TextStyle(  
        fontSize: 24,  
        color: Colors.white,  
    ),  
),  
(  
),  
),  
),  
Expanded(  
    child: Container(  
        color: Colors.red,  
        child: const Center(  
            child: Text(  
                '4',  
                style: TextStyle(  
                    fontSize: 24,  
                    color: Colors.white,  
                ),  
            ),  
        ),  
    ),  
),  
],  
(  
),  
),  
Expanded(  
    child: Row(  
        children: [  
            Expanded(  
                flex: 2,
```

```
        child: Container(  
            color: Colors.purple,  
            child: const Center(  
                child: Text(  
                    '5',  
                    style: TextStyle(  
                        fontSize: 24,  
                        color: Colors.white,  
                    ),  
                ),  
            ),  
        ),  
        Expanded(  
            child: Container(  
                color: Colors.blue,  
                child: const Center(  
                    child: Text(  
                        '6',  
                        style: TextStyle(  
                            fontSize: 24,  
                            color: Colors.white,  
                        ),  
                    ),  
                ),  
            ),  
        ),  
    ],  
,  
],
```



```
        ),  
        ),  
        ),  
        Expanded(  
            child: Container(  
                color: Colors.orange,  
                child: const Center(  
                    child: Text(  
                        '3',  
                        style: TextStyle(  
                            fontSize: 24,  
                            color: Colors.white,  
                        ),  
                    ),  
                ),  
            ),  
        ),  
        ),  
        ),  
        Expanded(  
            child: Container(  
                color: Colors.red,  
                child: const Center(  
                    child: Text(  
                        '4',  
                        style: TextStyle(  
                            fontSize: 24,  
                            color: Colors.white,  
                        ),  
                    ),  
                ),  
            ),  
        ),  
    ),  
),
```




Practical-8

Part – 8 (A) (Container with ScrollView)

Code:



Practical-8

Part – 8 (A) (Container with ScrollView)

Code:

```
import 'package:flutter/material.dart';

void main() {
  runApp(const ScrollApp());
}

class ScrollApp extends StatelessWidget {
  const ScrollApp({Key? key}) : super(key: key);

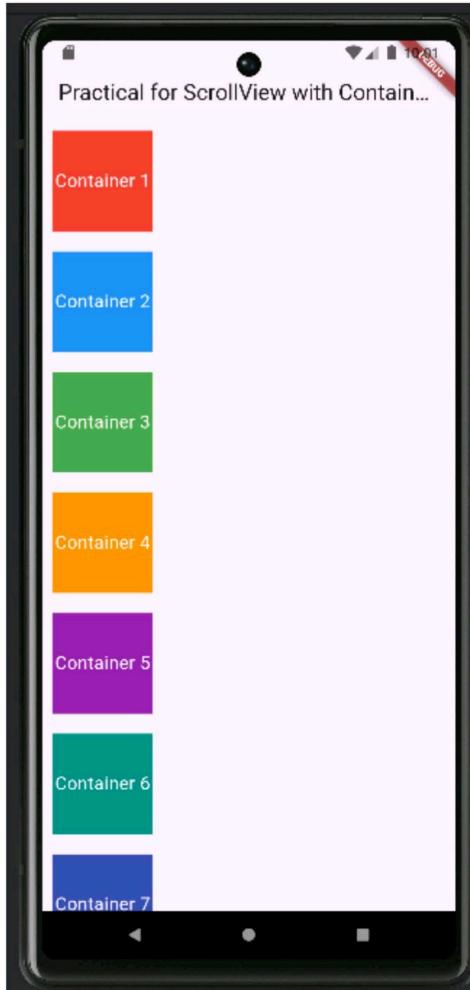
  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      title: 'ScrollView Example',
      home: const ScrollScreen(),
    );
  }
}

class ScrollScreen extends StatelessWidget {
  const ScrollScreen({Key? key}) : super(key: key);

  // Define a list of custom colors as a constant
  static const List<Color> customColors = [
    Colors.red,
    Colors.blue,
    Colors.green,
    Colors.orange,
    Colors.purple,
    Colors.teal,
    Colors.indigo,
    Colors.yellow,
  ]
}
```

```
Colors.brown,  
Colors.cyan,  
];  
  
@override  
Widget build(BuildContext context) {  
  return Scaffold(  
    appBar: AppBar(  
      title: const Text('Practical for ScrollView with Containers'),  
    ),  
    body: SingleChildScrollView(  
      child: Column(  
        children: List.generate(10, (index) {  
          return Container(  
            width: 100,  
            height: 100,  
            margin: const EdgeInsets.all(10),  
            color: customColors[index % customColors.length], // Use custom colors  
            child: Center(  
              child: Text(  
                'Container ${index + 1}',  
                style: const TextStyle(  
                  fontSize: 18,  
                  color: Colors.white,  
                ),  
              ),  
            ),  
          );  
        }),  
      ),  
    ),  
  );  
});  
},  
),
```

```
);  
}  
}
```



Part – 8 (B) (Custom font)

first download the font from google fonts

extract it to ‘fonts’ folder of the project

if ‘fonts’ not exist create it

change under main.dart

Code:

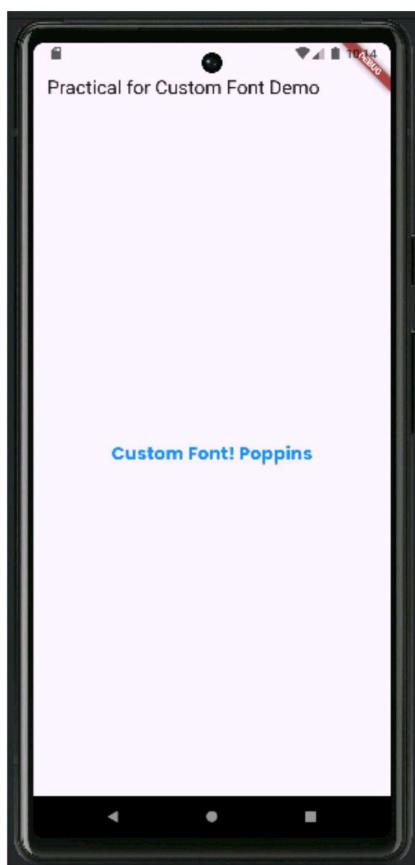
```
import 'package:flutter/material.dart';
```

```
void main() {
```

```
runApp(MyApp());  
}  
  
class MyApp extends StatelessWidget {  
  @override  
  Widget build(BuildContext context) {  
    return MaterialApp(  
      title: 'Custom Font Demo',  
      theme: ThemeData(  
        primarySwatch: Colors.blue,  
      ),  
      home: MyHomePage(),  
    );  
  }  
}
```

```
class MyHomePage extends StatelessWidget {  
  @override  
  Widget build(BuildContext context) {  
    return Scaffold(  
      appBar: AppBar(  
        title: Text('Practical for Custom Font Demo'),  
      ),  
      body: Center(  
        child: Text(  
          'Custom Font! '  
          'Poppins',  
          style: TextStyle(  
            color: Colors.blue,  
            fontFamily: 'Poppins',
```

```
fontSize: 20,  
fontWeight: FontWeight.bold,  
,  
,  
);  
}  
}  
  
pubspec.yaml  
  
flutter:  
fonts:  
- family: Poppins  
fonts:  
- asset: fonts/Poppins-Regular.ttf  
- asset: fonts/Poppins-Bold.ttf  
weight: 600
```



Practical-9e

Part – 9 (A) (Login page)

Code:

```
import 'package:flutter/material.dart';
```

```
void main() {
```

```
    runApp(MyApp());
```

```
}
```

```
class MyApp extends StatelessWidget {
```

```
    @override
```

```
    Widget build(BuildContext context) {
```

```
        return MaterialApp(
```

```
            title: 'Login App',
```

```
theme: ThemeData(  
    primarySwatch: Colors.yellow,  
,  
    home: LoginPage(),  
,  
)  
{  
}  
  
class LoginPage extends StatefulWidget {  
  
    @override  
    _LoginPageState createState() => _LoginPageState();  
}  
  
class _LoginPageState extends State<LoginPage> {  
    final TextEditingController _usernameController = TextEditingController();  
    final TextEditingController _passwordController = TextEditingController();  
    String _message = "  
";  
  
    void _login() {  
        // Replace with your actual username and password  
        const String correctUsername ='admin';  
        const String correctPassword = 'pass';  
  
        String username = _usernameController.text;  
        String password = _passwordController.text;  
  
        setState(() {  
            if (username == correctUsername && password == correctPassword) {  
                _message = 'Yayy!!Login Successful!';  
            } else {  
                _message = 'Sorry Login Failed! Incorrect username or password.';  
            }  
        });  
    }  
}
```

```
    }

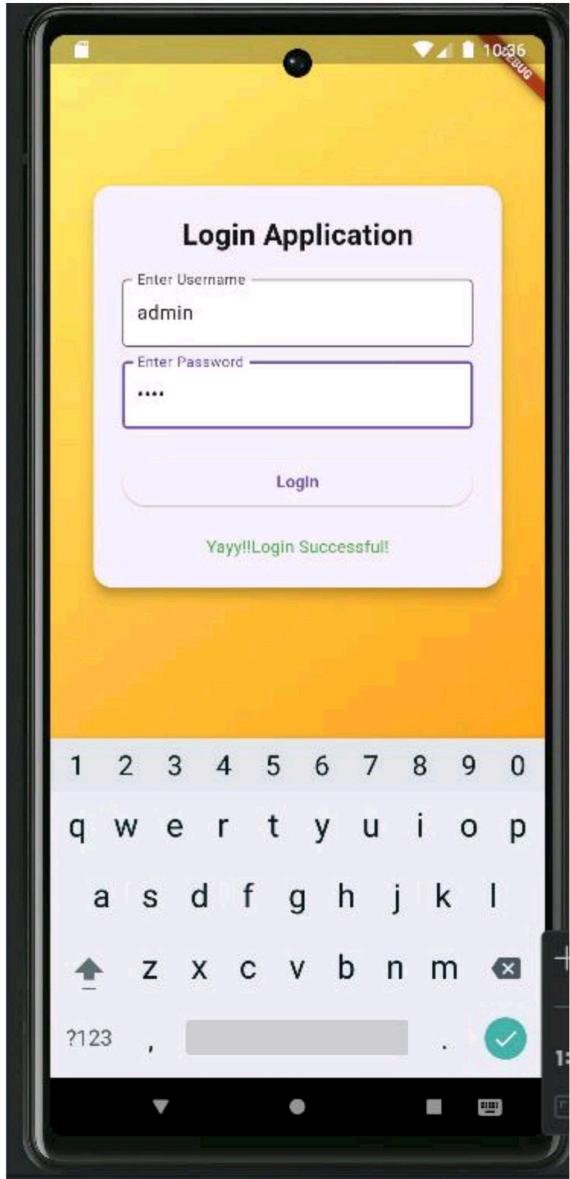
  });

}

@Override
Widget build(BuildContext context) {
  return Scaffold(
    body: Container(
      decoration: BoxDecoration(
        gradient: LinearGradient(
          colors: [Colors.yellow[300]!, Colors.yellow[800]!],
          begin: Alignment.topLeft,
          end: Alignment.bottomRight,
        ),
      ),
      child: Center(
        child: Padding(
          padding: const EdgeInsets.all(32.0),
          child: Card(
            elevation: 8,
            shape: RoundedRectangleBorder(
              borderRadius: BorderRadius.circular(16),
            ),
            child: Padding(
              padding: const EdgeInsets.all(24.0),
              child: Column(
                mainAxisAlignment: MainAxisAlignment.min,
                children: [
                  Text(
                    'Login Application',
                    style: TextStyle(fontSize: 24, fontWeight: FontWeight.bold),
                  ),
                ],
              ),
            ),
          ),
        ),
      ),
    ),
  );
}
```

```
),  
    SizedBox(height: 20),  
    TextField(  
        controller: _usernameController,  
        decoration: InputDecoration(  
            labelText: 'Enter Username',  
            border: OutlineInputBorder(),  
            filled: true,  
            fillColor: Colors.white,  
        ),  
,  
    SizedBox(height: 12),  
    TextField(  
        controller: _passwordController,  
        decoration: InputDecoration(  
            labelText: 'Enter Password',  
            border: OutlineInputBorder(),  
            filled: true,  
            fillColor: Colors.white,  
        ),  
        obscureText: true,  
,  
    SizedBox(height: 20),  
    ElevatedButton(  
        onPressed: _login,  
        child: Text('Login'),  
        style: ElevatedButton.styleFrom(  
            minimumSize: Size(double.infinity, 40),  
            padding: EdgeInsets.symmetric(horizontal: 16),  
        ),  
,
```

```
SizedBox(height: 20),  
Text(  
    _message,  
    style: TextStyle(  
        color: _message.startsWith('Yayy!!Login Successful!') ? Colors.green : Colors.red,  
    ),  
    ),  
    ],  
),  
),  
),  
),  
),  
),  
),  
),  
);  
}  
}
```



Part – 9 (B) (First -> second screen)

Code:

```
import 'package:flutter/material.dart';
```

```
void main() {  
  runApp(MyApp());  
}
```

```
class MyApp extends StatelessWidget {  
  @override  
  Widget build(BuildContext context) {
```

```
return MaterialApp(
  title: 'Data Pass App',
  theme: ThemeData(
    primarySwatch: Colors.yellow,
  ),
  home: FirstScreen(),
);

}

}

class FirstScreen extends StatelessWidget {
final TextEditingController _controller = TextEditingController();

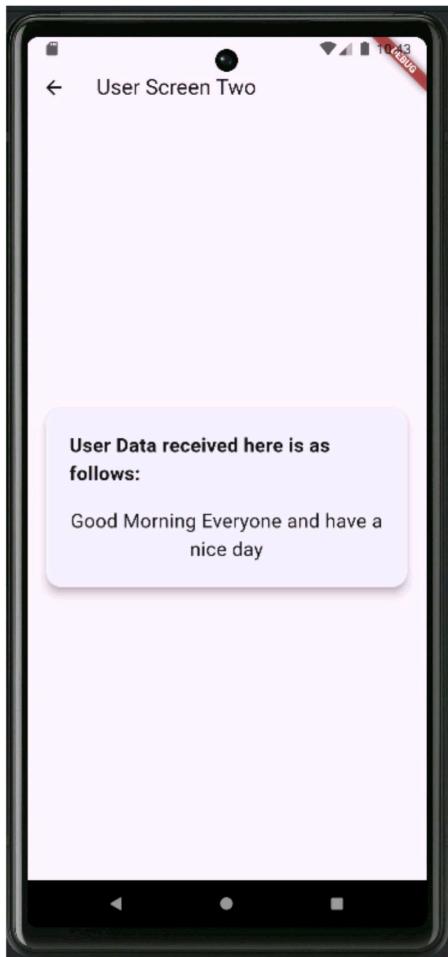
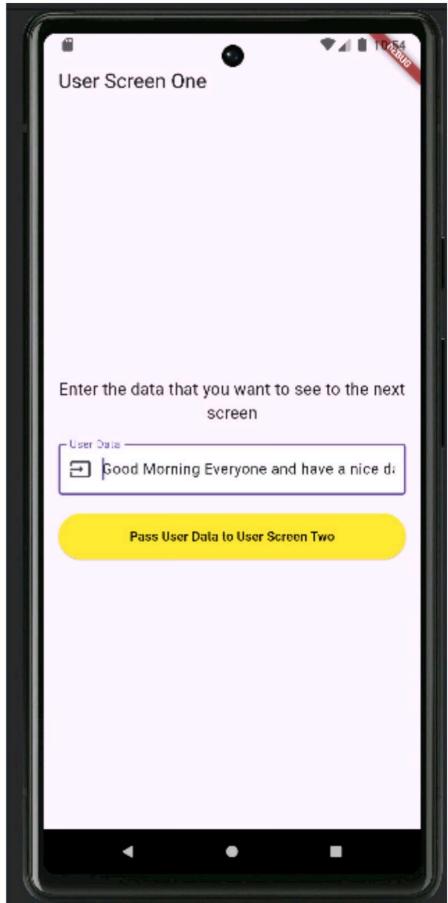
@Override
Widget build(BuildContext context) {
  return Scaffold(
    appBar: AppBar(
      title: Text('User Screen One'),
    ),
    body: Padding(
      padding: const EdgeInsets.all(16.0),
      child: Center(
        child: Column(
          mainAxisAlignment: MainAxisAlignment.center,
          children: [
            Text(
              'Enter the data that you want to see to the next screen',
              style: TextStyle(fontSize: 18),
              textAlign: TextAlign.center,
            ),
            SizedBox(height: 20),
          ],
        ),
      ),
    ),
  );
}
```

```
TextField(  
    controller: _controller,  
    decoration: InputDecoration(  
        labelText: 'User Data',  
        border: OutlineInputBorder(),  
        filled: true,  
        fillColor: Colors.white,  
        prefixIcon: Icon(Icons.input),  
    ),  
,  
SizedBox(height: 20),  
ElevatedButton(  
    onPressed: () {  
        if (_controller.text.isNotEmpty) {  
            Navigator.push(  
                context,  
                MaterialPageRoute(  
                    builder: (context) => SecondScreen(data: _controller.text),  
                ),  
            );  
        } else {  
            ScaffoldMessenger.of(context).showSnackBar(  
                SnackBar(content: Text('Please enter some data')),  
            );  
        }  
    },  
    child: Text('Pass User Data to User Screen Two'),  
    style: ElevatedButton.styleFrom(  
        backgroundColor: Colors.yellow, // Set button color to yellow  
        foregroundColor: Colors.black, // Set text color to black  
        minimumSize: Size(double.infinity, 50),
```

```
        ),  
        ),  
    ],  
    ),  
    ),  
    ),  
);  
}  
}  
  
class SecondScreen extends StatelessWidget {  
    final String data;  
  
    SecondScreen({required this.data});
```

```
    @override  
    Widget build(BuildContext context) {  
        return Scaffold(  
            appBar: AppBar(  
                title: Text('User Screen Two'),  
            ),  
            body: Center(  
                child: Padding(  
                    padding: const EdgeInsets.all(16.0),  
                    child: Card(  
                        elevation: 8,  
                        shape: RoundedRectangleBorder(  
                            borderRadius: BorderRadius.circular(16),  
                        ),  
                        child: Padding(  
                            padding: const EdgeInsets.all(24.0),  
                        ),  
                    ),  
                ),  
            ),  
        );  
    }  
}
```

```
child: Column(  
    mainAxisSize: MainAxisSize.min,  
    children: [  
        Text(  
            'User Data received here is as follows:',  
            style: TextStyle(fontSize: 20, fontWeight: FontWeight.bold),  
        ),  
        SizedBox(height: 20),  
        Text(  
            data,  
            style: TextStyle(fontSize: 20),  
            textAlign: TextAlign.center,  
        ),  
    ],  
,  
,  
,  
,  
,  
);  
}  
}
```



Practical-10

Part – 10 (Hero Animation)

change in main.dart

Code:

```
import 'package:flutter/material.dart';
```

```
void main() {
```

```
    runApp(MyApp());
```

```
}
```

```
class MyApp extends StatelessWidget {
```

```
    @override
```

```
    Widget build(BuildContext context) {
```

```
        return MaterialApp(
```

```
title: 'Practical for Hero Animation',  
theme: ThemeData(  
    primarySwatch: Colors.yellow,  
,  
home: FirstPage(),  
);  
}  
}
```

```
class FirstPage extends StatelessWidget {  
@override  
Widget build(BuildContext context) {  
return Scaffold(  
    appBar: AppBar(  
        title: Text('Practical for Hero Animation'),  
,  
    body: Center(  
        child: GestureDetector(  
            onTap: () {  
                Navigator.of(context).push(  
                    MaterialPageRoute(  
                        builder: (context) => SecondPage(),  
,  
                );  
            },  
        ),  
        child: Hero(  
            tag: 'hero-image',  
            child: Image.asset(  
                'assets/image2.jpg',  
                width: 100,  
                height: 100,  
            ),  
        ),  
    ),  
);  
}
```

```
        ),  
        ),  
        ),  
        ),  
    );  
}  
}
```

```
class SecondPage extends StatelessWidget {
```

```
    @override  
    Widget build(BuildContext context) {  
        return Scaffold(  
            appBar: AppBar(  
                title: Text('Hero Animation!!!'),  
            ),  
            body: Center(  
                child: Hero(  
                    tag: 'hero-image',  
                    child: Image.asset('assets/image3.jpg'),  
                ),  
            ),  
        );  
    }  
}
```

```
pubsec.yaml file
```

```
flutter:  
    uses-material-design: true
```

```
assets:
```

- assets/image1.jpg
- assets/image2.jpg

- assets/image3.jpg



